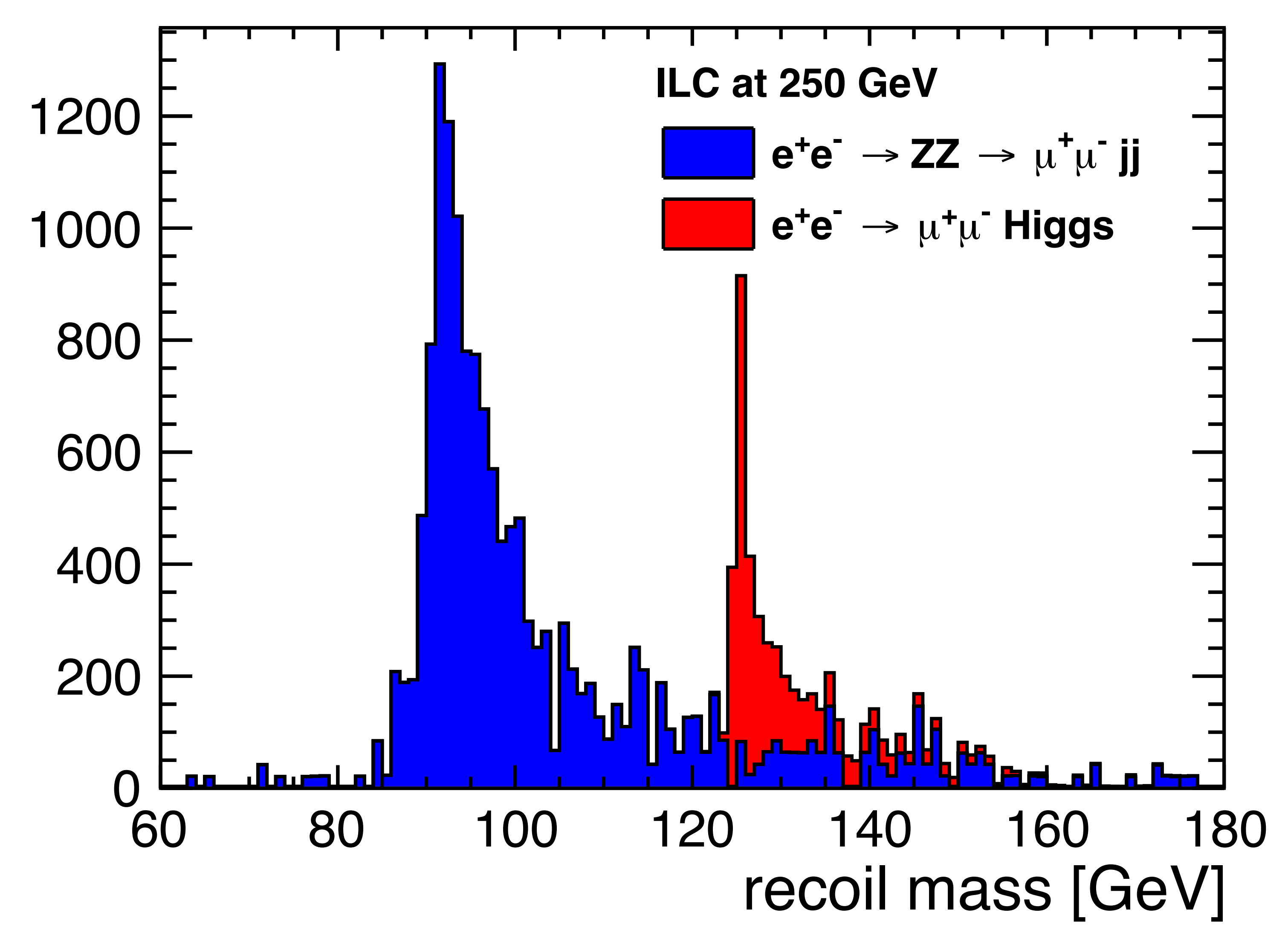


Your first ILC analysis: Higgs recoil mass from miniDST via root

J. List (DESY) on behalf of the LCC Physics & Software Working Groups

Snowmass EF Tutorial Aug 28, 2020

Before we start

- please raise your zoom hand for any category that you consider a match!

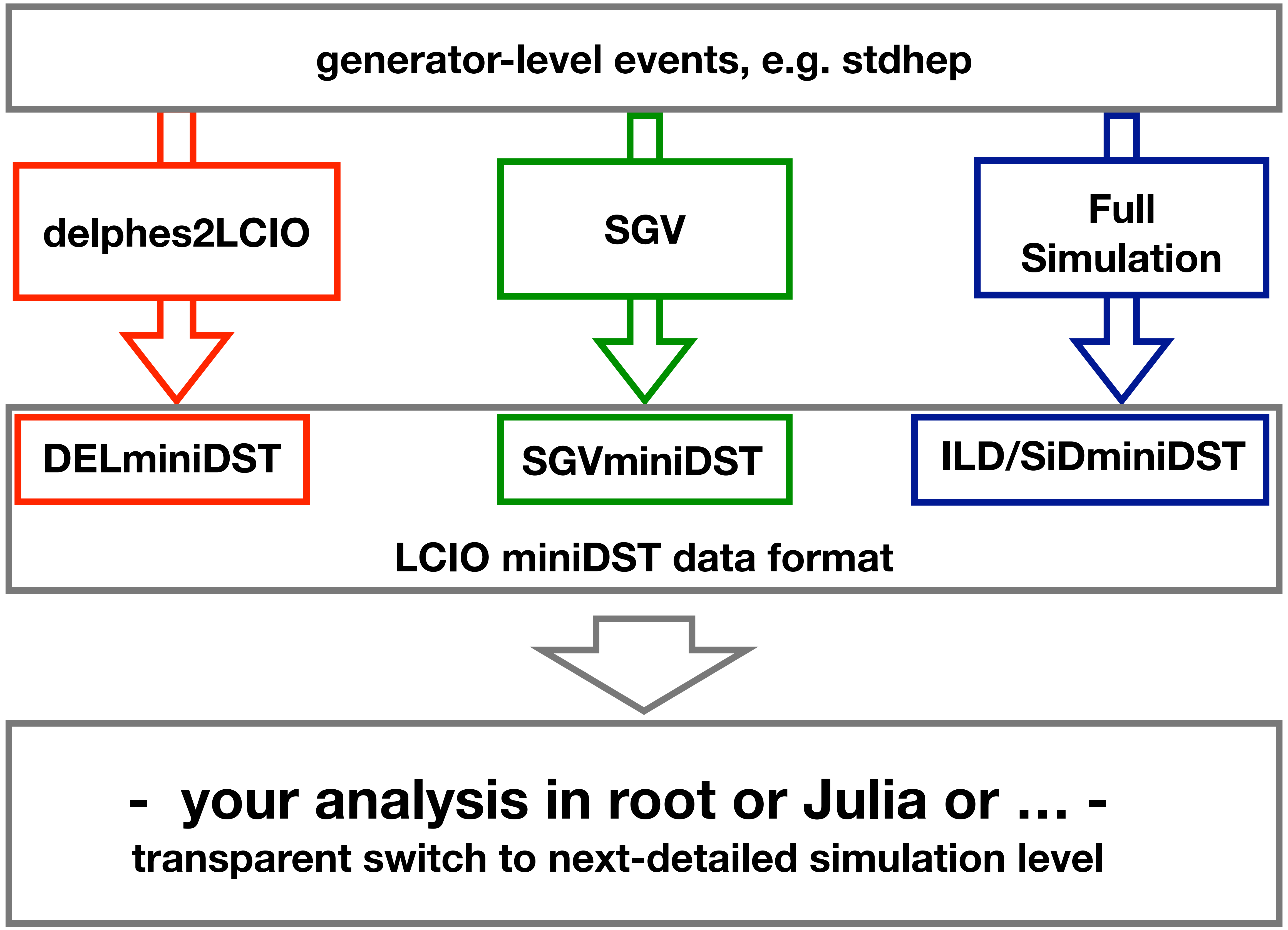
- are you:
  - an undergraduate student?
  - a post-graduate student?
  - a postdoc?
  - a senior scientist / professor?

- have you ever worked on a (past/present/future)
  - e+e- experiment?
  - pp experiment?
  - other particle physics experiments?
  - particle physics theory?

- why are you here:
  - you plan to do an e^+e^- study for Snowmass
    - have already topic in mind
    - still looking for a topic
  - just curious, but no concrete plan to do a Snowmass study
  - co-organiser / convener

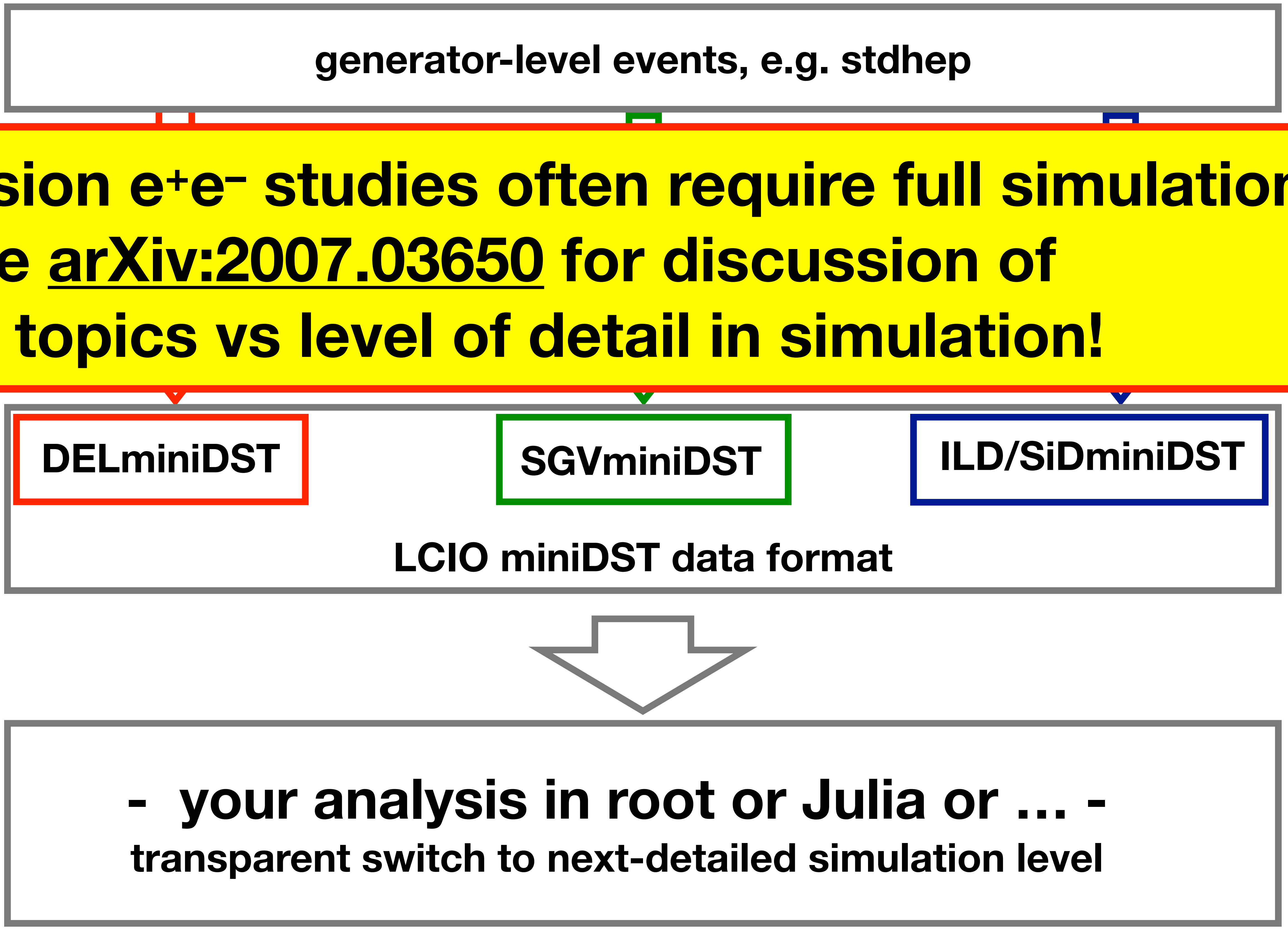
Parametrised, fast and full (=geant4-based) simulations

- **delphes2lcio**: an lcio application which makes Delphes (parametrised detector simulation) write out LCIO (https://github.com/iLCSoft/LCIO/tree/master/examples/cpp/delphes2lcio)
- **SGV**: **S**imulation a **G**rande **V**itesse (https://www.desy.de/~berggren/sgv_ug/sgv_ug.html) - detailed fast simulation from `first principles” (nearly no parametrisations!)
- **iLCSoft** (https://github.com/iLCSoft): software suite for full simulation and reconstruction of ILC & CLIC detectors



Parametrised, fast and full (=geant4-based) simulations

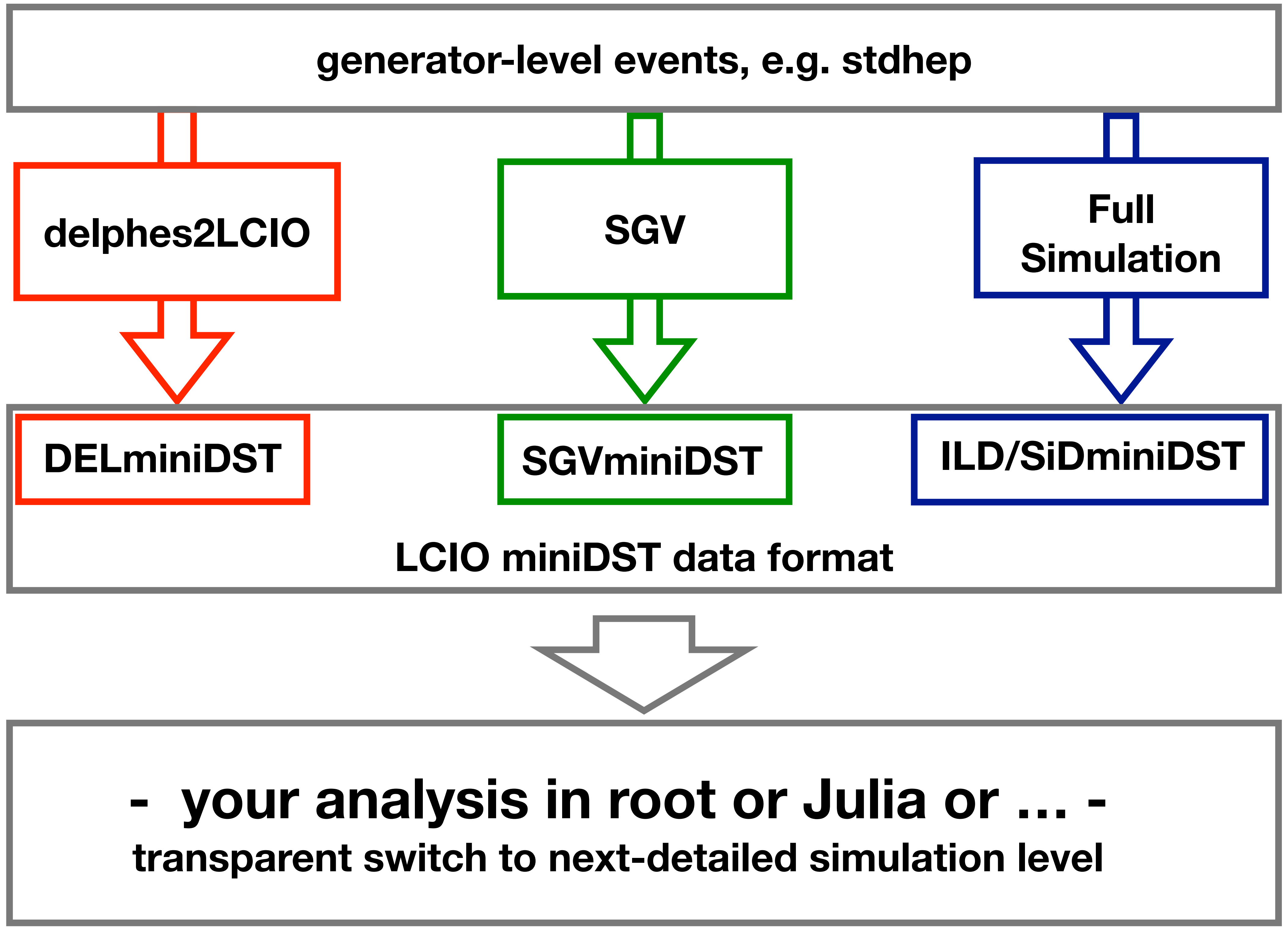
- **delphes2lcio**: an lcio application which makes Delphes (parametrised detector simulation) write out LCIO (https://github.com/iLCSoft/LCIO master/examples/cpp/delphes2lc
- **SGV**: **S**imulation a **G**rande **V**itesse (https://www.desy.de/~berggren/sgv_ug/sgv_ug.html) - detailed fast simulation from ``first principles" (nearly no parametrisations!)
- **iLCSoft** (https://github.com/iLCSoft): software suite for full simulation and reconstruction of ILC & CLIC detectors



NOTE: precision e^+e^- studies often require full simulation - see arXiv:2007.03650 for discussion of study topics vs level of detail in simulation!

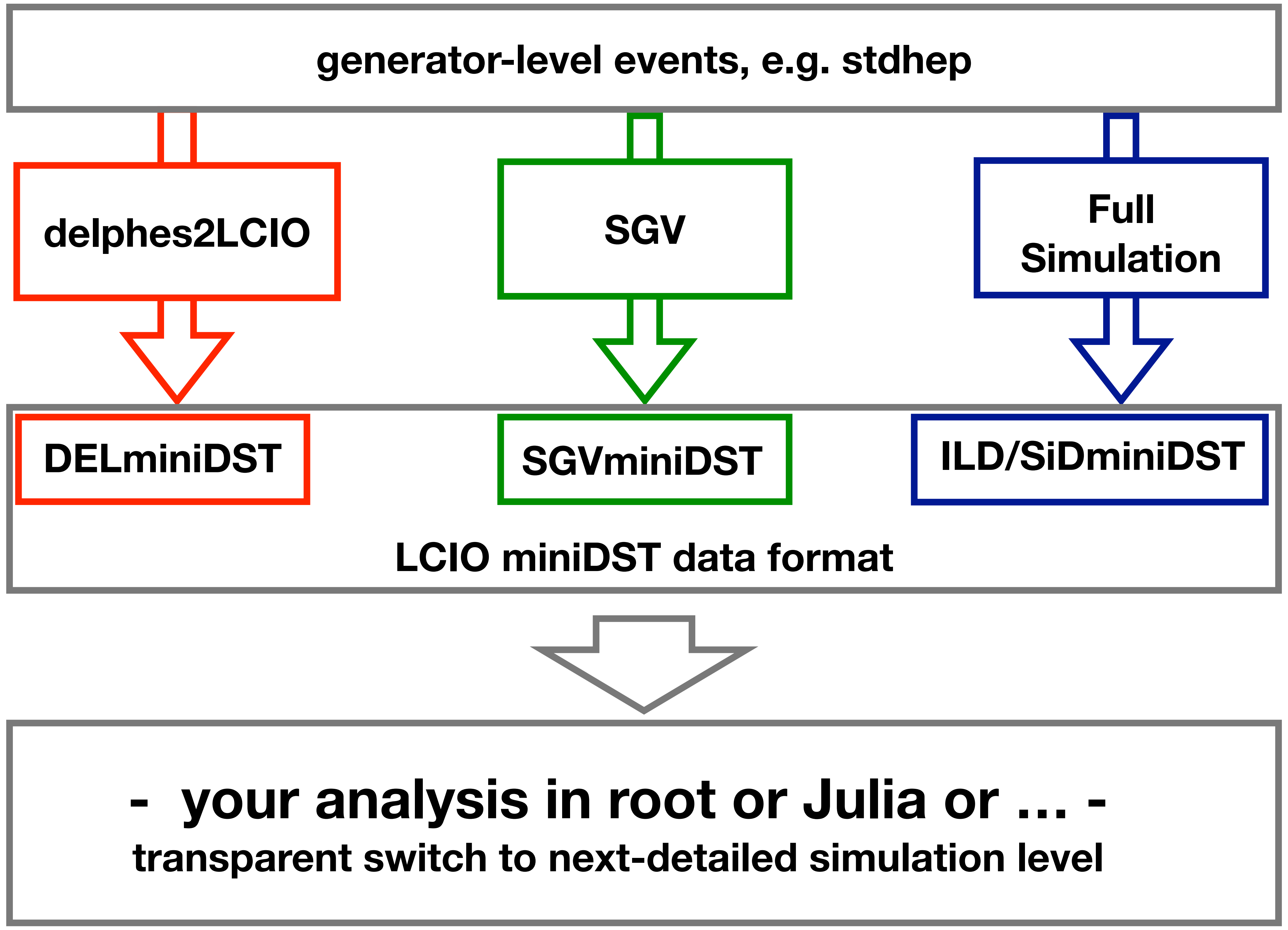
LCIO & miniDST

- **LCIO** (**L**inear **C**ollider **I/O**, part of iLCSoft, https://github.com/iLCSoft/LCIO):
  - event data model and persistency framework
  - implemented for C++, Fortran, Java, Go, Python => will see root and jupyter examples today!
- **miniDST,** https://github.com/ILDAnaSoft/miniDST:
  - high-level LCIO-file containing information very similar to Delphes root tree
  - can be filled from Delphes, SGV and full simulation
- ***analyses based on miniDST can easily switch between parametrised, fast and full simulation!***



LCIO & miniDST

- **LCIO** (**L**inear **C**ollider **I/O**, part of iLCSoft, https://github.com/iLCSoft/LCIO):
  - event data model and persistency framework
  - implemented for C++, Fortran, Java, Go, Python => will see root and jupyter examples today!
- **miniDST,** https://github.com/ILDAnaSoft/miniDST:
  - high-level LCIO-file containing information very similar to Delphes root tree
  - can be filled from Delphes, SGV and full simulation
- **analyses based on miniDST can easily switch between parametrised, fast and full simulation!**



LCIO is *not* ILC-specifc -
it's used by all future e^+e^- colliders
in one way or the other!

The miniDST format

COLLECTION NAME (SGV / ILD full sim)	COLLECTION NAME (Delphes)	EXPLANATION
PandoraPFOs	PFOs	particle flow objects from the main detector, incl. event shape variables
IsolatedElectrons	IsolatedElectrons	
IsolatedMuons	IsolatedMuons	
IsolatedTaus	IsolatedTaus	
IsolatedPhotons	IsolatedPhotons	
Refined2Jets	Durham2Jets	PandoraPFOs minus "IsolatedX" forced into 2 jets (Durham algorithm, plus flavour tag)
Refined3Jets	Durham3Jets	PandoraPFOs minus "IsolatedX" forced into 3 jets (Durham algorithm, plus flavour tag)
Refined4Jets	Durham4Jets	PandoraPFOs minus "IsolatedX" forced into 4 jets (Durham algorithm, plus flavour tag)
Refined5Jets	Durham5Jets	PandoraPFOs minus "IsolatedX" forced into 5 jets (Durham algorithm, plus flavour tag)
Refined6Jets	Durham6Jets	PandoraPFOs minus "IsolatedX" forced into 6 jets (Durham algorithm, plus flavour tag)
BCalPFOs	N/A	particle flow objects from the most forward calorimeter
PrimaryVertex	N/A	
PrimaryVertex_RP	N/A	"reconstructed particle" representing the primary vertex
MCParticlesSkimmed	MCParticles	
MCTruthRecoLink	MCTruthRecoLink	links from MCParticles to PandoraPFOs
RecoMCTruthLink	RecoMCTruthLink	links from PandoraPFOs to MCParticles

Getting started

- open https://github.com/ILDAnaSoft/miniDST in your browser -
  all the following parts of the tutorial are described there!
- log in to your working computer, e.g. the OSG:
  `ssh -Y [username]@login.snowmass21.io`
- use bash shell:
  `bash`
- download the examples:
  `git clone https://github.com/ILDAnaSoft/miniDST.git`
- change to miniDST folder:
  `cd miniDST`
- set up versions of root, cmake, gcc, python etc:
  `.  setenv4LCIO.sh`
- next: download and install LCIO => next slide

Installing LCIO (incl. delphes2lcio)

- go to https://github.com/iLCSoft/LCIO
- go one directory up: `cd ..`
- download LCIO:
  `git clone https://github.com/iLCSoft/LCIO.git`
- change to LCIO folder, create and change to build directory:
  `cd LCIO; mkdir build; cd build`
- run cmake with option to build root dictionaries and C++17:
  `cmake -DBUILD_ROOTDICT=ON -D CMAKE_CXX_STANDARD=17 ..`
- make it:
  `make -j 4 install`
- test your installation:
  `make test`
- set up paths etc:
  `cd ..; . ./setup.sh`

Installing LCIO (incl. delphes2lcio)

- go to https://github.com/iLCSoft/LCIO
- go one directory up: `cd ..`
- download LCIO:
  `git clone https://github.com/iLCSoft/LCIO.git`
- change to LCIO folder, create and change to build directory:
  `cd LCIO; mkdir build; cd build`
- run cmake with option to build root dictionaries and C++17:
  `cmake -DBUILD_ROOTDICT=ON -D CMAKE_CXX_STANDARD=17 ..`
- make it:
  `make -j 4 install`
- test your installation:
  `make test`
- set up paths etc:
  `cd ..; . ./setup.sh`

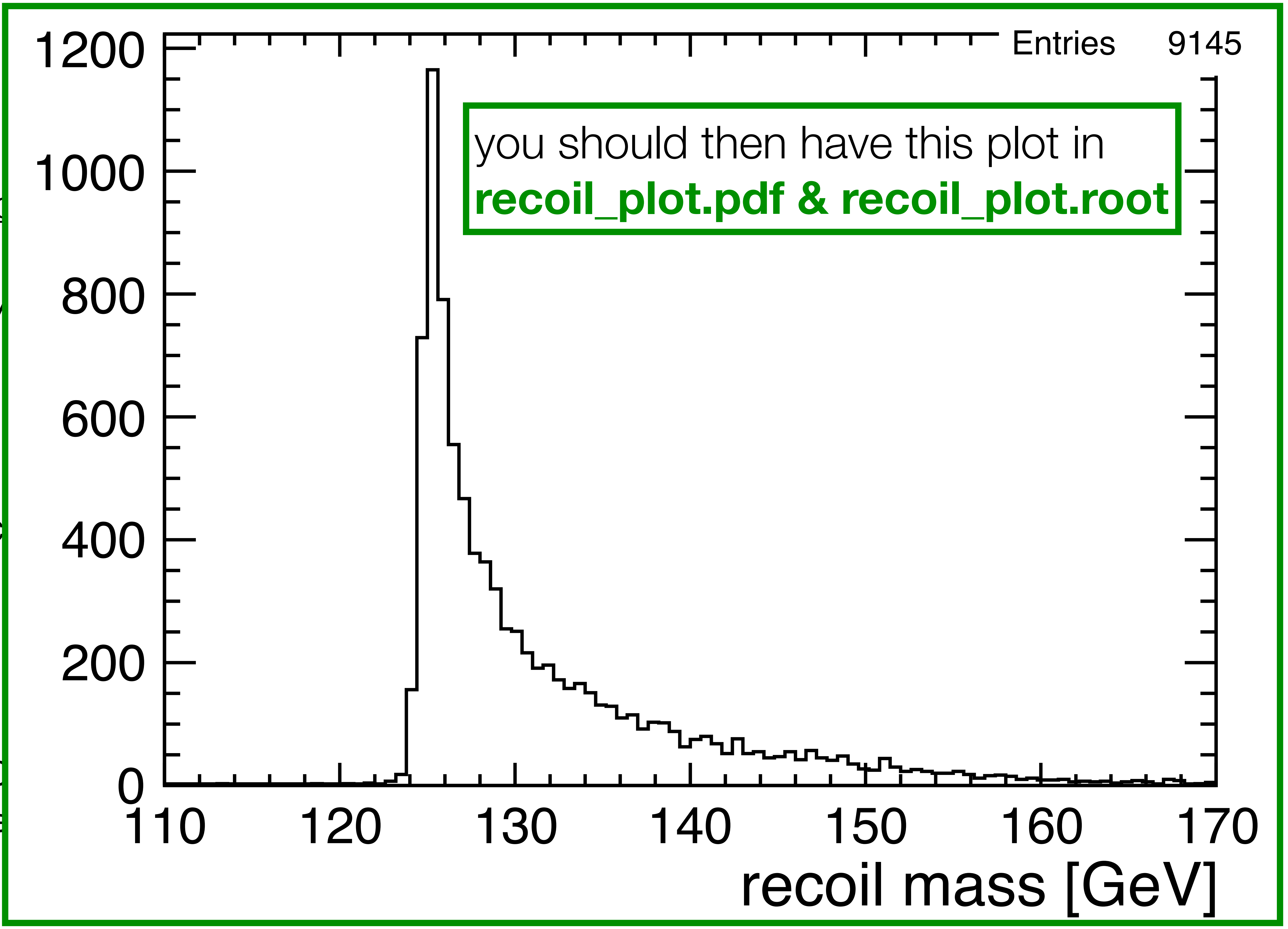
Hint: If you have (rather recent) gcc, python, root and cmake on your laptop / local desktop, try to install LCIO there for more convenient interactive work in root!

The first plot

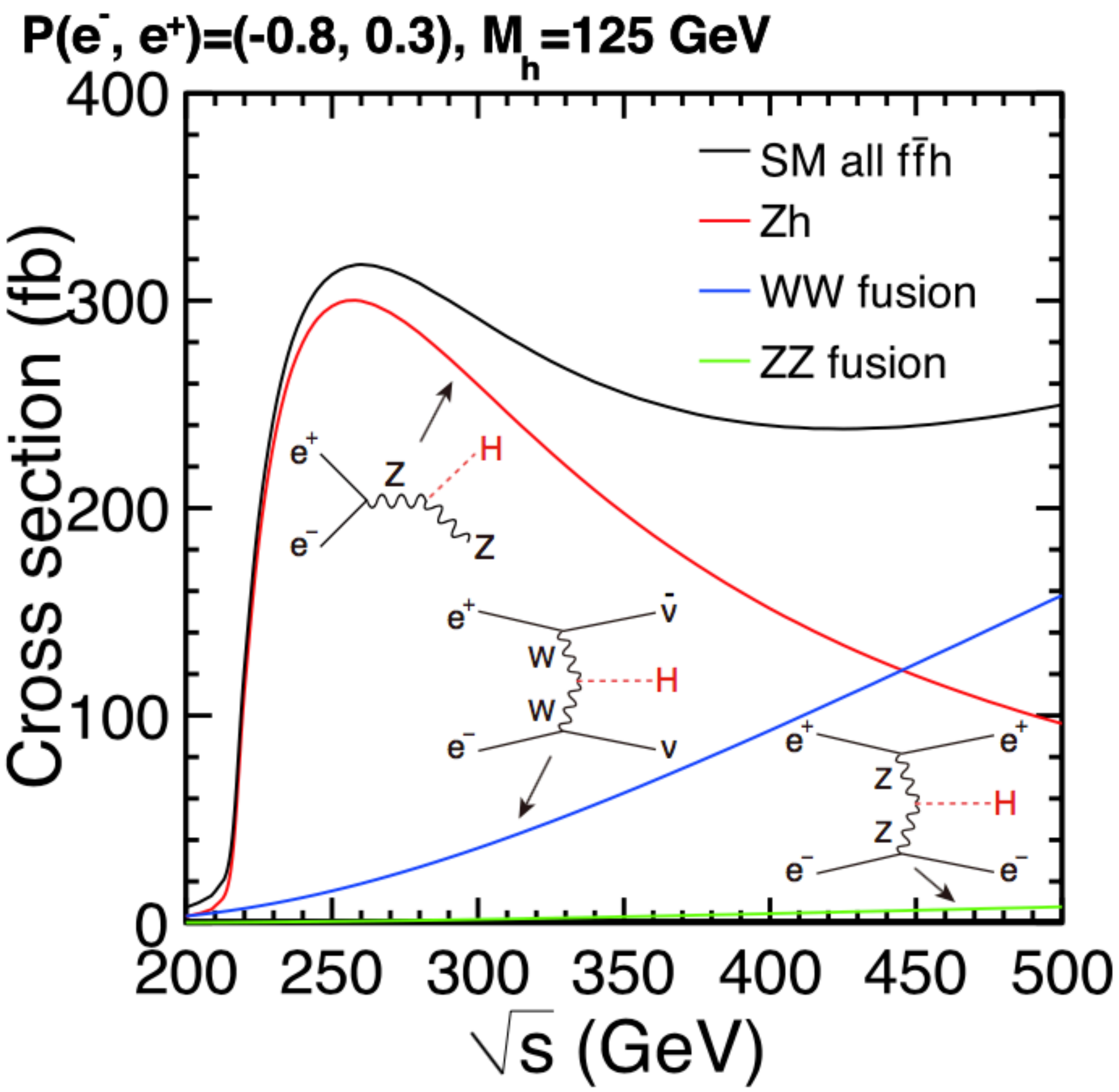
- go to back to miniDST/examples:
  `cd ../miniDST/examples`
- get the data:
  - either: download from links given on https://github.com/ILDAnaSoft/miniDST
  - or link to existing copy on OSG:
    `ln -s /collab/project/snowmass21/data/ilc/tutorial data`
- start root:
  `root -b`   (on your local computer, also -l)
- this reads the provided .rootlogon.C
  => most of it is optional and a matter of taste - the crucial part is:
  gInterpreter->AddIncludePath(“$LCIO");
  gSystem->Load("${LCIO}/lib/liblcio.so");
  gystem->Load(“${LCIO}/lib/liblcioDict.so");
- run first macro:
  ```
  .x higgs_recoil.C (“data/rv01-16-p10_250.sv01-14-01-p00.mILD_o1_v05.E250-TDR_ws.I106479.Pe2e2h.eL.pR-00001-ILDminiDST.slcio”);
  ```
- quit root:
  `.q`

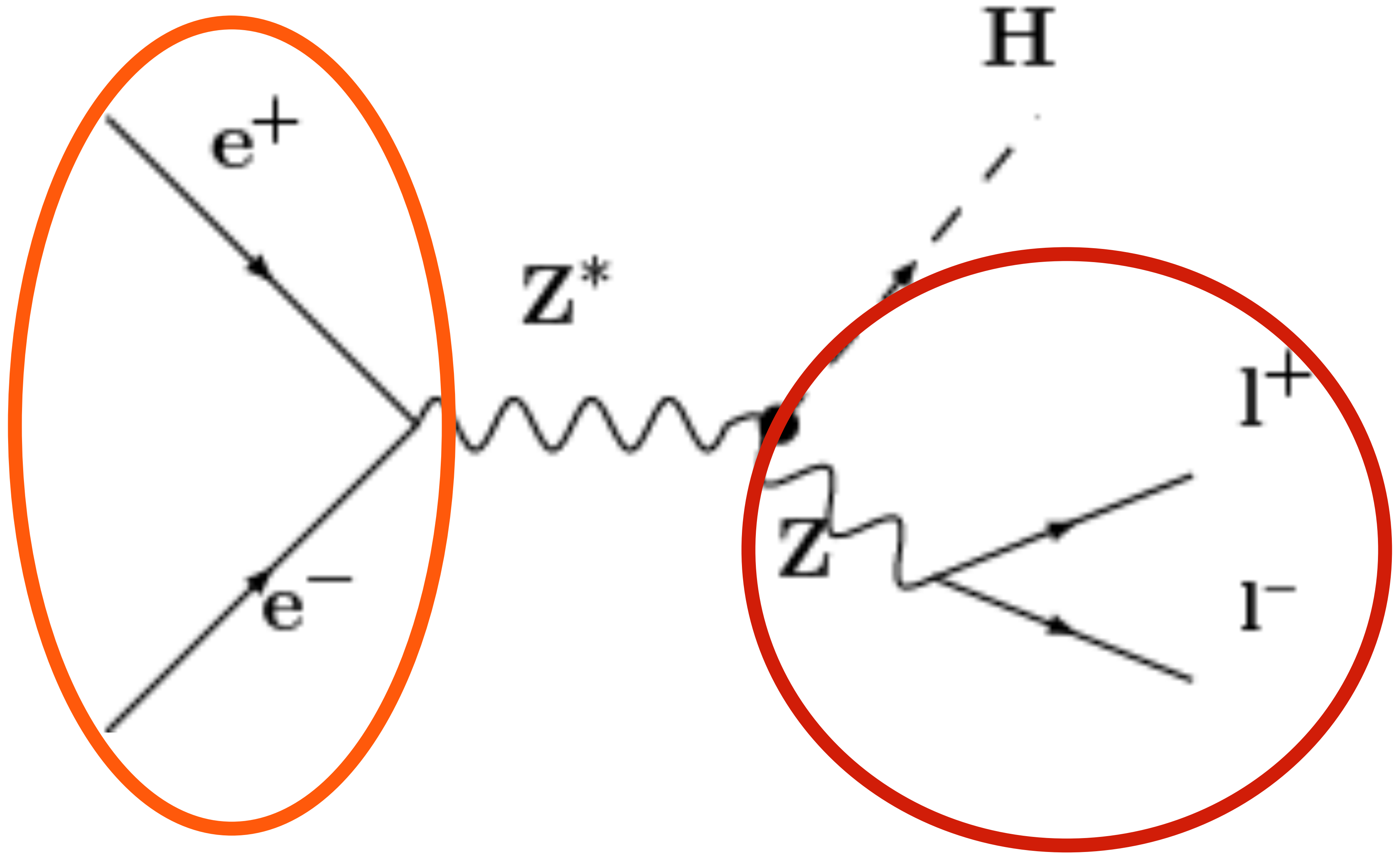
The first plot

- go to back to miniDST/examples:
  `cd ../miniDST/examples`
- get the data:
  - either: download from links given on https://github.co
  - or link to existing copy on OSG:
    `ln -s /collab/project/snowmass21/data/`
- start root:
  `root -b`   (on your local computer, also -l)
- this reads the provided .rootlogon.C
  => most of it is optional and a matter of taste - the cruc
  gInterpreter->AddIncludePath(“$LCIO");
  gSystem->Load("${LCIO}/lib/liblcio.so");
  gystem->Load(“${LCIO}/lib/liblcioDict.so");
- run first macro:
  ```
  .x higgs_recoil.C (“data/rv01-16-p10_250
  p00.mILD_o1_v05.E250-TDR_ws.I106479.Pe2e
  ILDminiDST.slcio”);
  ```
- quit root:
  `.q`



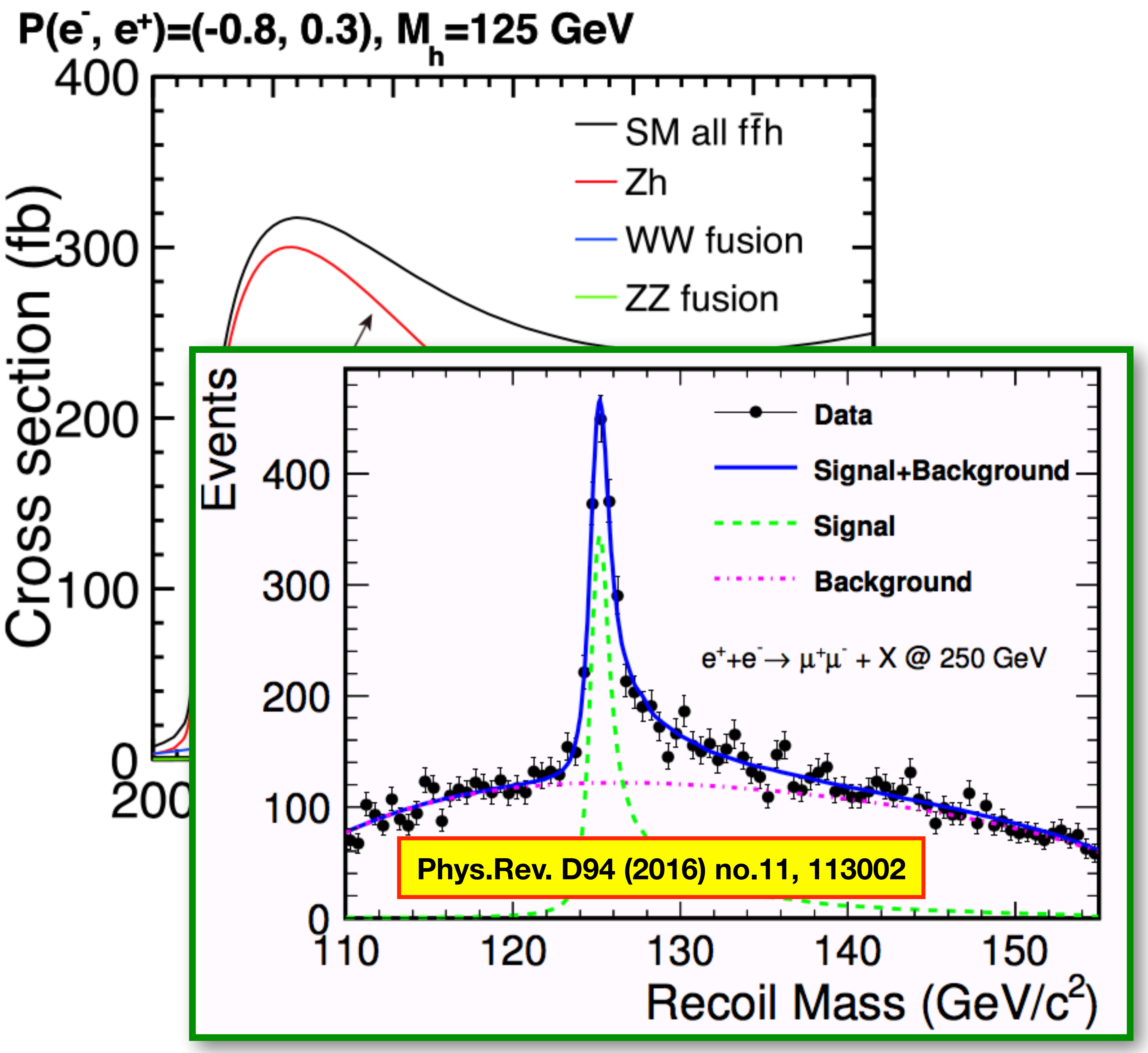
What are we seeing?

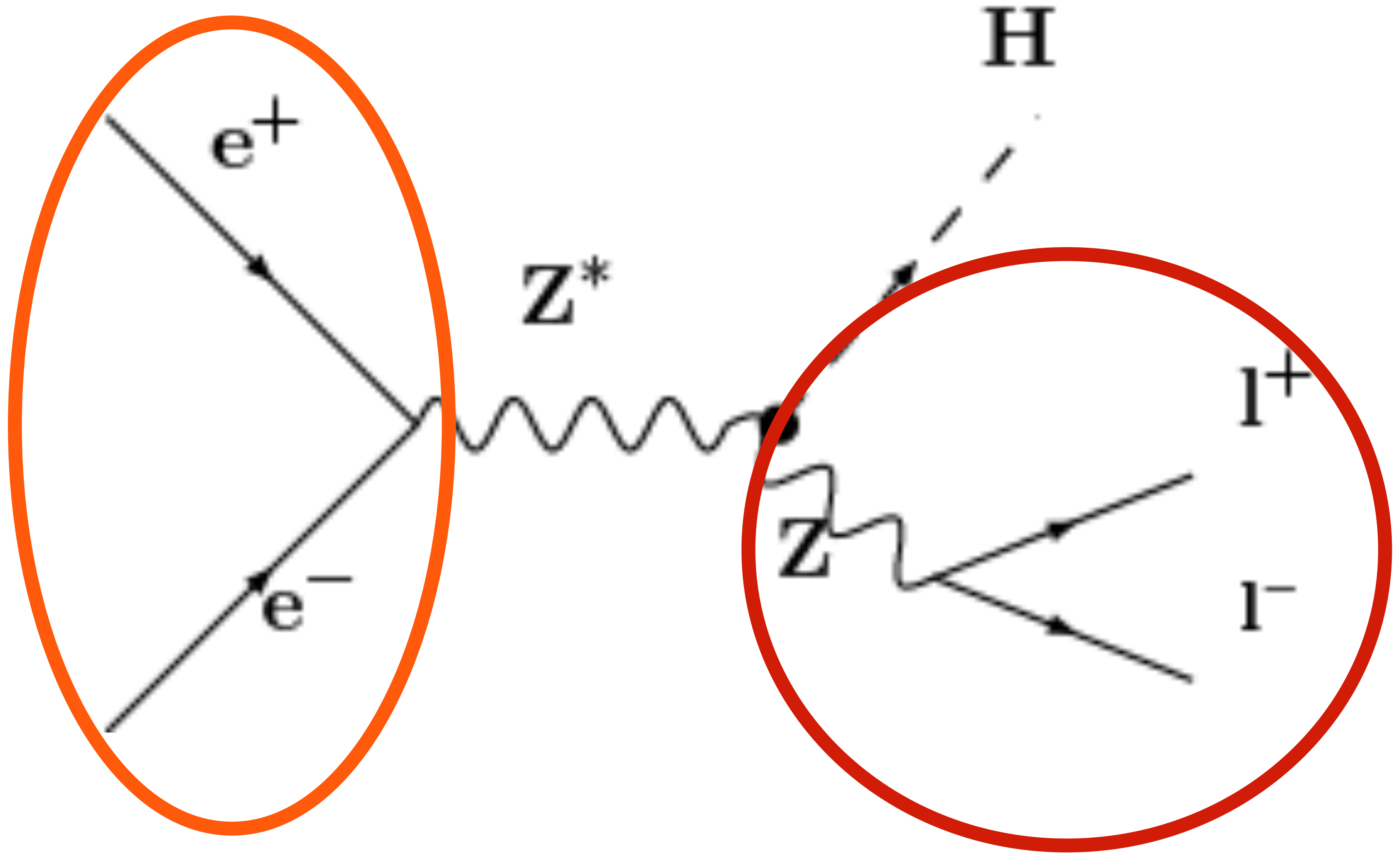




$$M_H^2 = M_{recoil}^2 = s + M_Z^2 - 2E_Z\sqrt{s}$$

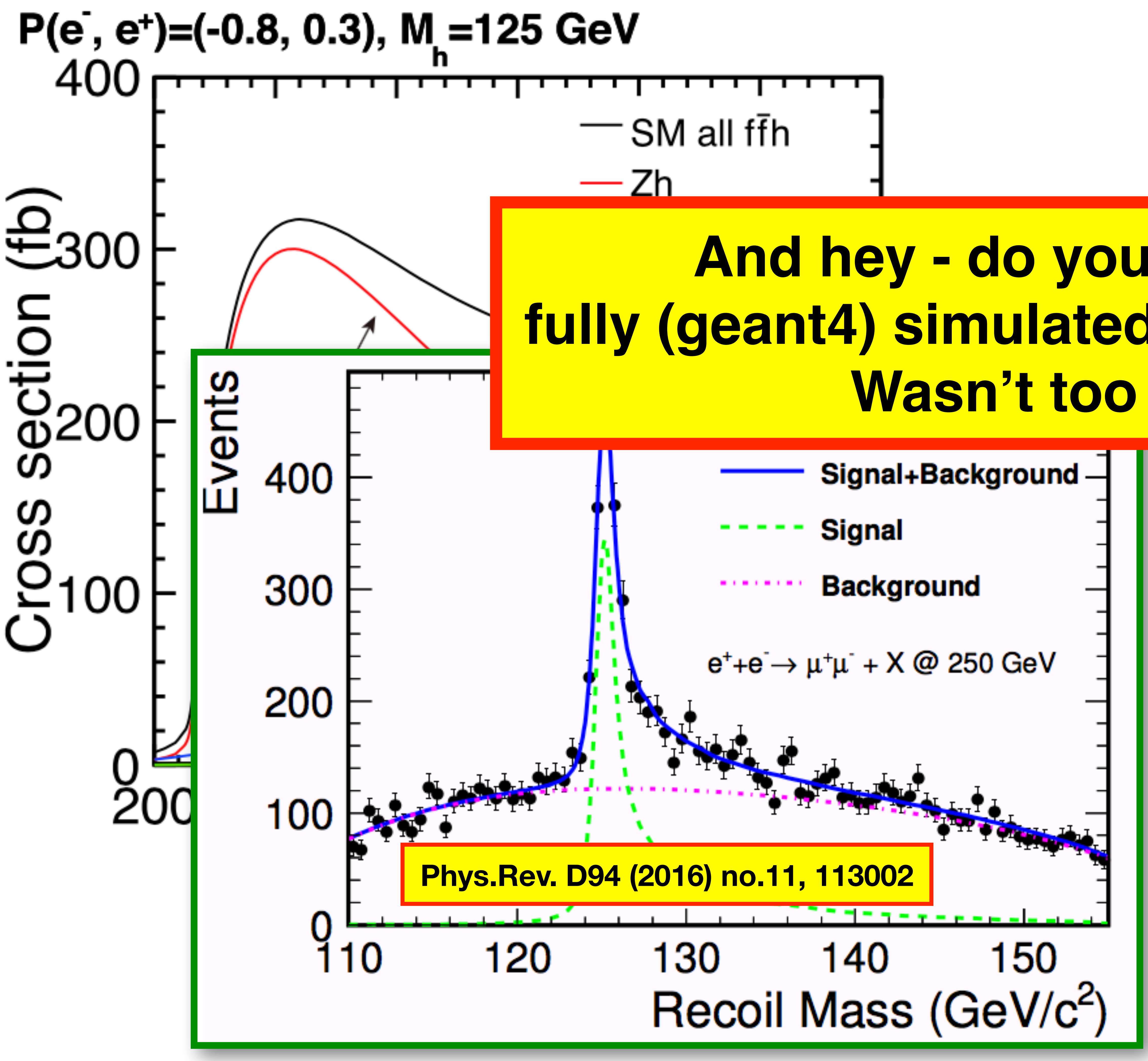
What are we seeing?



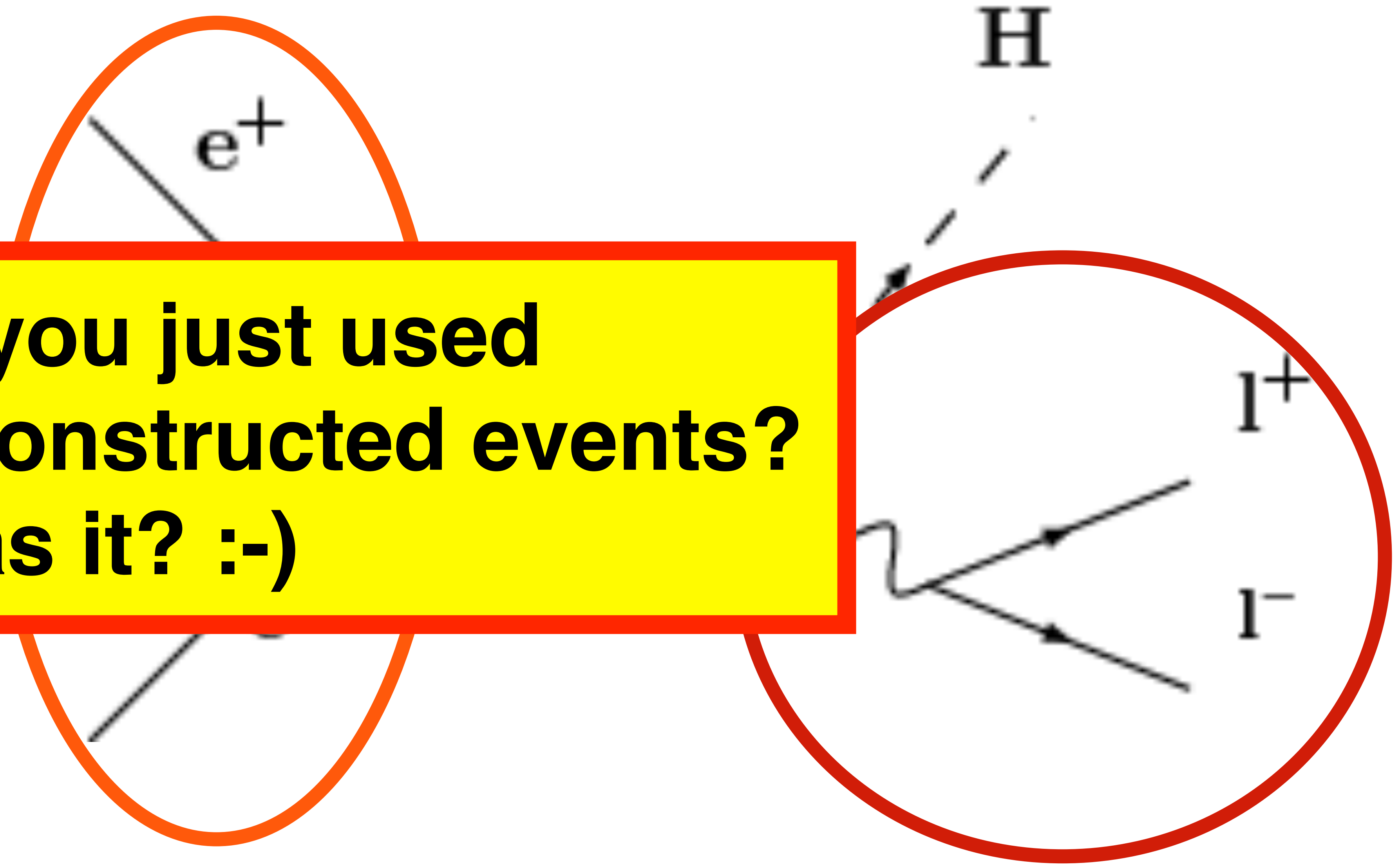


$$M_H^2 = M_{recoil}^2 = s + M_Z^2 - 2E_Z\sqrt{s}$$

What are we seeing?

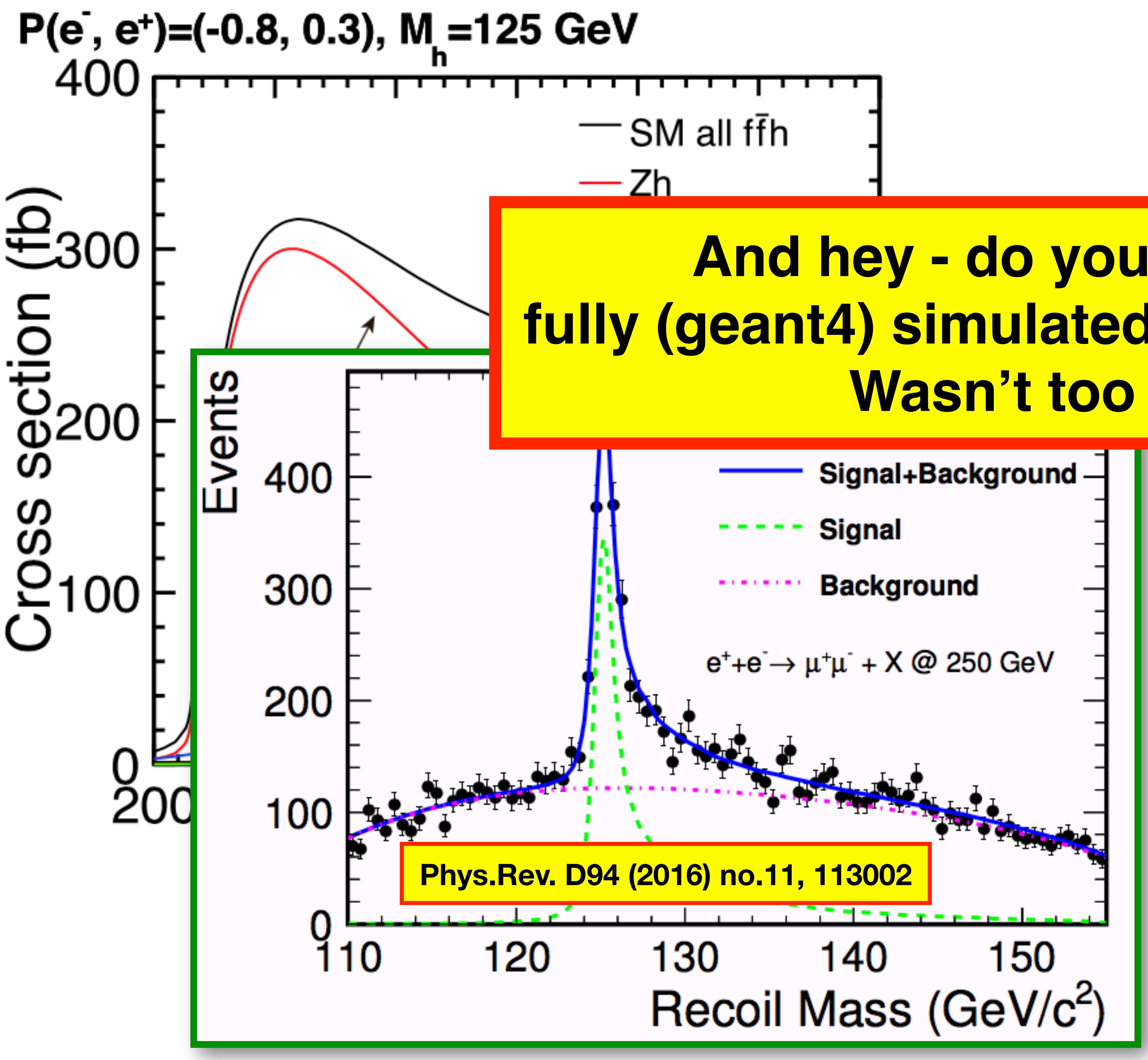


And hey - do you realise you just used fully (geant4) simulated and reconstructed events? Wasn't too hard, was it? :-)

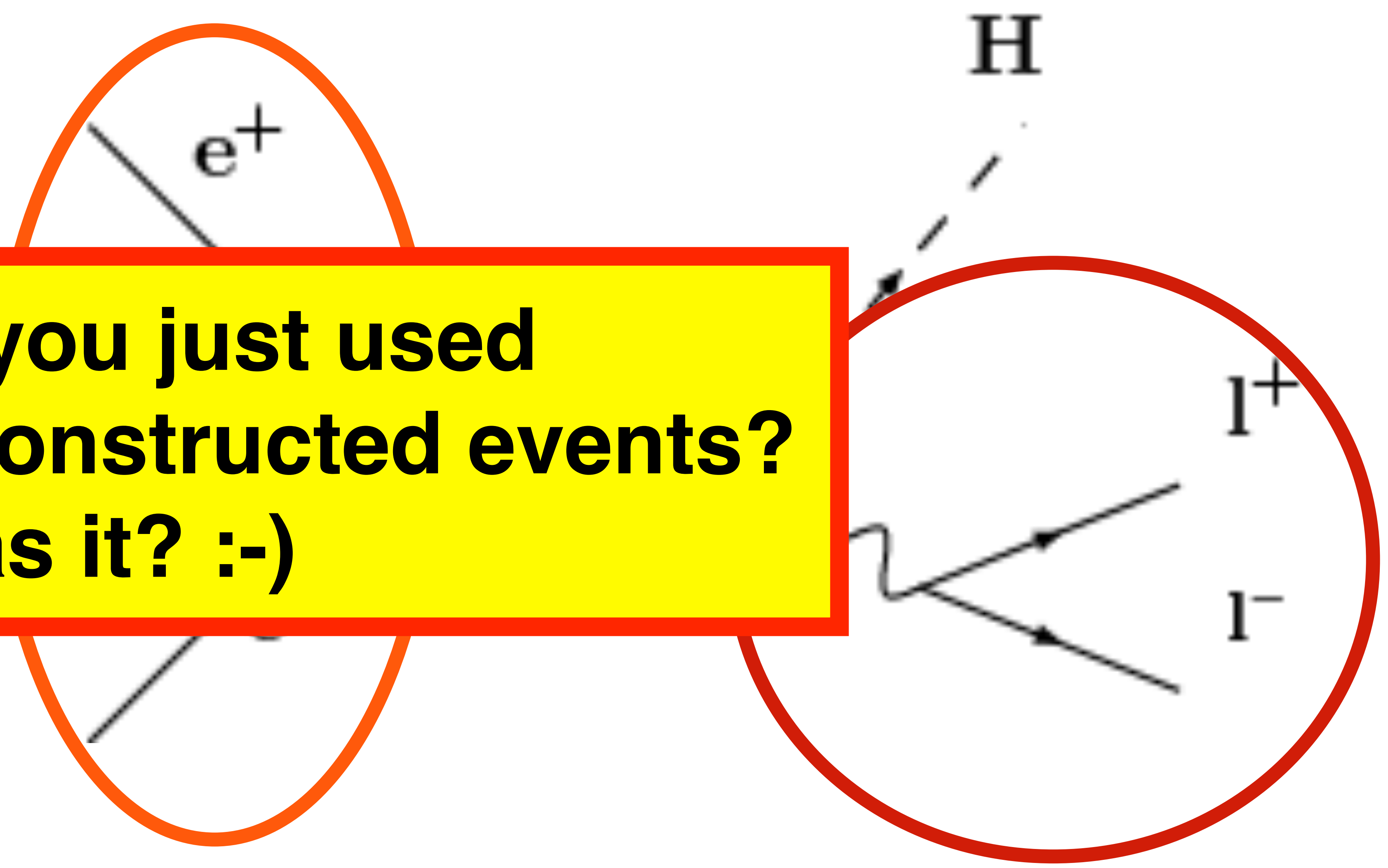


$$M_H^2 = M_{recoil}^2 = s + M_Z^2 - 2E_Z\sqrt{s}$$

What are we seeing?



And hey - do you realise you just used fully (geant4) simulated and reconstructed events? Wasn't too hard, was it? :-)



$$M_H^2 = M_{recoil}^2 = s + M_Z^2 - 2E_Z\sqrt{s}$$

Let's add some backgrounds…

Interlude 1: nomenclature of processes

- more details c.f. https://indico.cern.ch/event/868940/contributions/3814465/
- classify physics processes by the number of (fermions + antifermions) in the final state:
  - ee -> 2f: ee -> f fbar   (f = e, mu, tau, u, d, s, c, b, nu)
  - ee -> 4f: mostly WW / ZZ, but taking into account ***all*** contributing matrix elements and their interference
  - ee -> 6f: mostly ttbar, some ZZZ, WWZ, but again, all MEs + interference considered!
  - "SM" samples: mass of Higgs is set to huge value, so that Feynman diagrams containing the Higgs are ***not*** included
  - instead have separately: ee -> ffbar h
- file name contains:

rv01-16-p10_250.sv01-14-01-p00.mILD_o1_v05.E250-TDR_ws.I106479.Pe2e2h.eL.pR-00001-ILDminiDST.slcio

Interlude 1: nomenclature of processes

- more details c.f. https://indico.cern.ch/event/868940/contributions/3814465/
- classify physics processes by the number of (fermions + antifermions) in the final state:
  - ee -> 2f: ee -> f fbar   (f = e, mu, tau, u, d, s, c, b, nu)
  - ee -> 4f: mostly WW / ZZ, but taking into account ***all*** contributing matrix elements and their interference
  - ee -> 6f: mostly ttbar, some ZZZ, WWZ, but again, all MEs + interference considered!
  - "SM" samples: mass of Higgs is set to huge value, so that Feynman diagrams containing the Higgs are ***not*** included
  - instead have separately: ee -> ffbar h
- file name contains:

rv01-16-p10_250.sv01-14-01-p00.mILD_o1_v05.E250-TDR_ws.I106479.Pe2e2h.eL.pR-00001-ILDminiDST.slcio

process:
e2 = muon, so
mumu h
(I dentification number)

Interlude 1: nomenclature of processes

- more details c.f. https://indico.cern.ch/event/868940/contributions/3814465/
- classify physics processes by the number of (fermions + antifermions) in the final state:
  - ee -> 2f: ee -> f fbar   (f = e, mu, tau, u, d, s, c, b, nu)
  - ee -> 4f: mostly WW / ZZ, but taking into account ***all*** contributing matrix elements and their interference
  - ee -> 6f: mostly ttbar, some ZZZ, WWZ, but again, all MEs + interference considered!
  - "SM" samples: mass of Higgs is set to huge value, so that Feynman diagrams containing the Higgs are ***not*** included
  - instead have separately: ee -> ffbar h
- file name contains:

polarisation:
electron L (eft-handed)
positron R (ight-handed)

rv01-16-p10_250.sv01-14-01-p00.mILD_o1_v05.E250-TDR_ws.I106479.Pe2e2h.eL.pR-00001-ILDminiDST.slcio

process:
e2 = muon, so
mumu h
(I dentification number)

Interlude 1: nomenclature of processes

- more details c.f. https://indico.cern.ch/event/868940/contributions/3814465/
- classify physics processes by the number of (fermions + antifermions) in the final state:
  - ee -> 2f: ee -> f fbar (f = e, mu, tau, u, d, s, c, b, nu)
  - ee -> 4f: mostly WW / ZZ, but taking into account ***all*** contributing matrix elements and their interference
  - ee -> 6f: mostly ttbar, some ZZZ, WWZ, but again, all MEs + interference considered!
  - "SM" samples: mass of Higgs is set to huge value, so that Feynman diagrams containing the Higgs are ***not*** included
  - instead have separately: ee -> ffbar h
- file name contains:

polarisation:
electron L (eft-handed)
positron R (ight-handed)

rv01-16-p10_250.sv01-14-01-p00.mILD_o1_v05.E250-TDR_ws.I106479.Pe2e2h.eL.pR-00001-ILDminiDST.slcio

process:
e2 = muon, so
mumu h
(I dentification number)

serial file number
& file format

Interlude 1: nomenclature of processes

- more details c.f. https://indico.cern.ch/event/868940/contributions/3814465/
- classify physics processes by the number of (fermions + antifermions) in the final state:
  - ee -> 2f: ee -> f fbar (f = e, mu, tau, u, d, s, c, b, nu)
  - ee -> 4f: mostly WW / ZZ, but taking into account ***all*** contributing matrix elements and their interference
  - ee -> 6f: mostly ttbar, some ZZZ, WWZ, but again, all MEs + interference considered!
  - "SM" samples: mass of Higgs is set to huge value, so that Feynman diagrams containing the Higgs are ***not*** included
  - instead have separately: ee -> ffbar h
- file name contains:

polarisation:
electron L (eft-handed)
positron R (ight-handed)

rv01-16-p10_250.sv01-14-01-p00.mILD_o1_v05.E250-TDR_ws.I106479.Pe2e2h.eL.pR-00001-ILDminiDST.slcio

energy and beam parameters: 250 GeV, TDR_ws

process: e2 = muon, so mumu h

(I dentification number)

serial file number & file format

Interlude 1: nomenclature of processes

- more details c.f. https://indico.cern.ch/event/868940/contributions/3814465/
- classify physics processes by the number of (fermions + antifermions) in the final state:
  - ee -> 2f: ee -> f fbar   (f = e, mu, tau, u, d, s, c, b, nu)
  - ee -> 4f: mostly WW / ZZ, but taking into account ***all*** contributing matrix elements and their interference
  - ee -> 6f: mostly ttbar, some ZZZ, WWZ, but again, all MEs + interference considered!
  - “SM” samples: mass of Higgs is set to huge value, so that Feynman diagrams containing the Higgs are ***not*** included
  - instead have separately: ee -> ffbar h
- file name contains:

detector model

polarisation:
electron L (eft-handed)
positron R (ight-handed)

rv01-16-p10_250.sv01-14-01-p00.mILD_o1_v05.E250-TDR_ws.I106479.Pe2e2h.eL.pR-00001-ILDminiDST.slcio

energy and
beam parameters:
250 GeV,
TDR_ws

process:
e2 = muon, so
mumu h

(I dentification number)

serial file number
& file format

Interlude 1: nomenclature of processes

- more details c.f. https://indico.cern.ch/event/868940/contributions/3814465/
- classify physics processes by the number of (fermions + antifermions) in the final state:
  - ee -> 2f: ee -> f fbar   (f = e, mu, tau, u, d, s, c, b, nu)
  - ee -> 4f: mostly WW / ZZ, but taking into account ***all*** contributing matrix elements and their interference
  - ee -> 6f: mostly ttbar, some ZZZ, WWZ, but again, all MEs + interference considered!
  - "SM" samples: mass of Higgs is set to huge value, so that Feynman diagrams containing the Higgs are ***not*** included
  - instead have separately: ee -> ffbar h
- file name contains:

polarisation:
electron L (eft-handed)
positron R (ight-handed)

detector model

rv01-16-p10_250.sv01-14-01-p00.mILD_o1_v05.E250-TDR_ws.I106479.Pe2e2h.eL.pR-00001-ILDminiDST.slcio

iLCSoft versions for reconstruction and simulation

energy and beam parameters: 250 GeV, TDR_ws

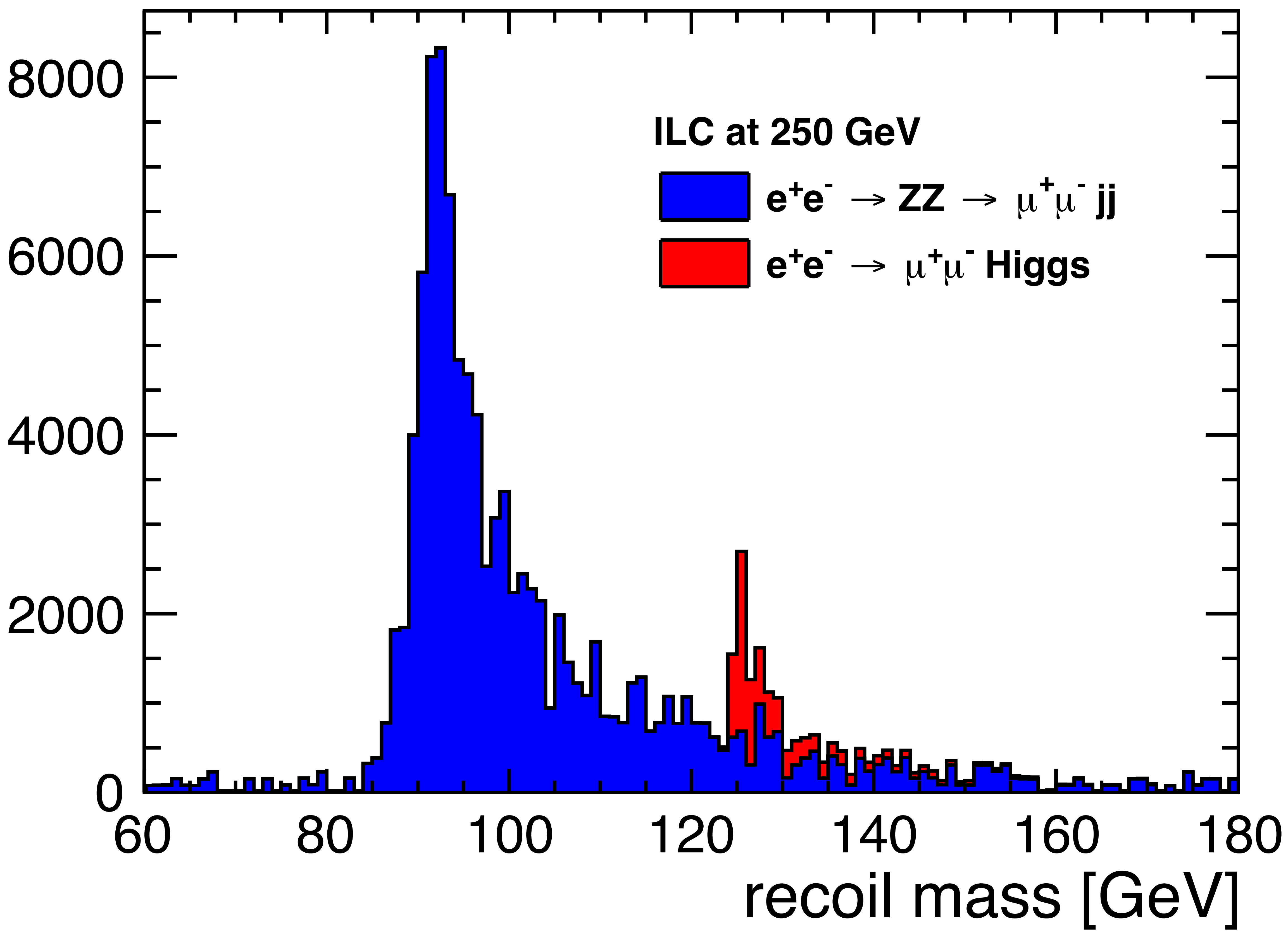
process: e2 = muon, so mumu h

(I dentification number)

serial file number & file format

And now with background

- restart root:
  `root -b`
- `.x higgs_recoil_with_bkg.C ("data/");`
- you should then have this plot in
  - **recoil_plot_with_bkg.pdf & recoil_plot_with_bkg.root**
- macro takes as further (optional) arguments:
  - double lumi_target=900., // 900 fb-1
  - double epol_target=-0.8, // P(e-) =-80%
  - double ppol_target=+0.3. // P(e+)=+30%
- try to change these settings to the opposite polarisation signs and redo the plot!



Interlude 2: Why these funny values?
- The ILC Strawman Running Scenario & Polarisation

all up-to-date numbers in ILC input document to the European strategy

- beam polarisation absolute values:
  - Electron beam: $|P(e^-)| \geq 80\%$
  - Positron beam: $|P(e^+)| = 30\%$,
    at 500 GeV upgradable to 60%
    at 1 TeV assume 20%
- **Notation: $(P(e^-) , P(e^+))$**
- sharing of luminosity between polarisation signs:

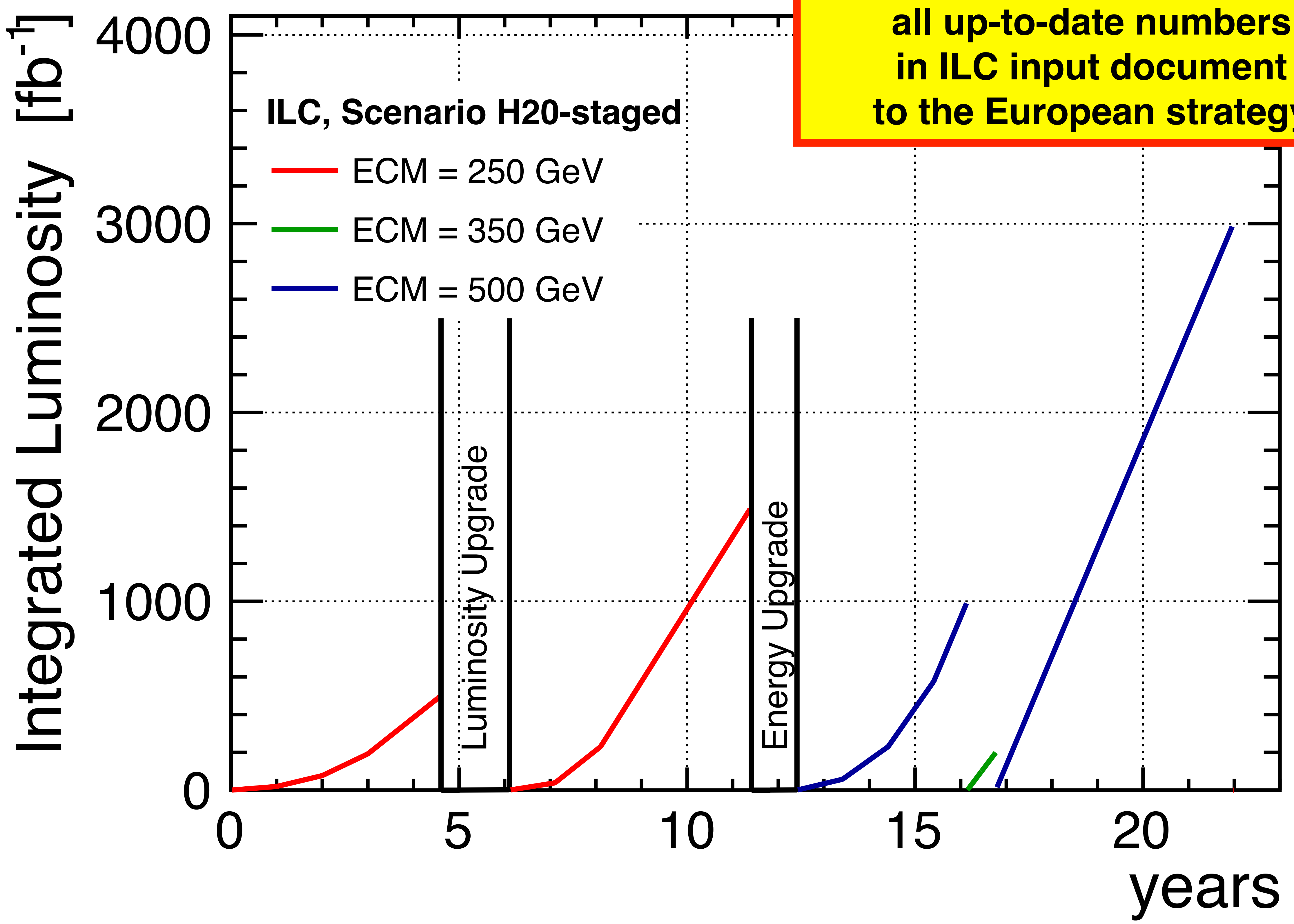
$\sqrt{s}$	$\int \mathscr{L} dt$	-+	+-	++	- -
250 GeV	2 ab^{-1}	$\boxed{0.9\ ab^{-1}}$	0.9 ab^{-1}	0.1 ab^{-1}	0.1 ab^{-1}
350 GeV	200 fb^{-1}	135 fb^{-1}	45 fb^{-1}	10 fb^{-1}	10 fb^{-1}
500 GeV	4 ab^{-1}	1.6 ab^{-1}	1.6 ab^{-1}	0.4 ab^{-1}	0.4 ab^{-1}
1 TeV	8 ab^{-1}	3.2 ab^{-1}	3.2 ab^{-1}	0.8 ab^{-1}	0.8 ab^{-1}
91 GeV	100 fb^{-1}	40 fb^{-1}	40 fb^{-1}	10 fb^{-1}	10 fb^{-1}
161 GeV	500 fb^{-1}	340 fb^{-1}	110 fb^{-1}	25 fb^{-1}	25 fb^{-1}

detailed reasoning c.f. arXiv:1506.07830

Interlude 2: Why these funny values?
- The ILC Strawman Running Scenario & Polarisation

- beam polarisation absolute values:
  - Electron beam: $|P(e^-)| \geq 80\%$
  - Positron beam: $|P(e^+)| = 30\%$,
    at 500 GeV upgradable to 60%
    at 1 TeV assume 20%
- **Notation: $(\,P(e^-)\,,\,P(e^+)\,)$**
- sharing of luminosity between polarisation signs:

√s	∫ℒdt	-+	+-	++	- -
250 GeV	2 ab^{-1}	$\boxed{0.9\ ab^{-1}}$	0.9 ab^{-1}	0.1 ab^{-1}	0.1 ab^{-1}
350 GeV	200 fb^{-1}	135 fb^{-1}	45 fb^{-1}	10 fb^{-1}	10 fb^{-1}
500 GeV	4 ab^{-1}	1.6 ab^{-1}	1.6 ab^{-1}	0.4 ab^{-1}	0.4 ab^{-1}
1 TeV	8 ab^{-1}	3.2 ab^{-1}	3.2 ab^{-1}	0.8 ab^{-1}	0.8 ab^{-1}
91 GeV	100 fb^{-1}	40 fb^{-1}	40 fb^{-1}	10 fb^{-1}	10 fb^{-1}
161 GeV	500 fb^{-1}	340 fb^{-1}	110 fb^{-1}	25 fb^{-1}	25 fb^{-1}



all up-to-date numbers in ILC input document to the European strategy

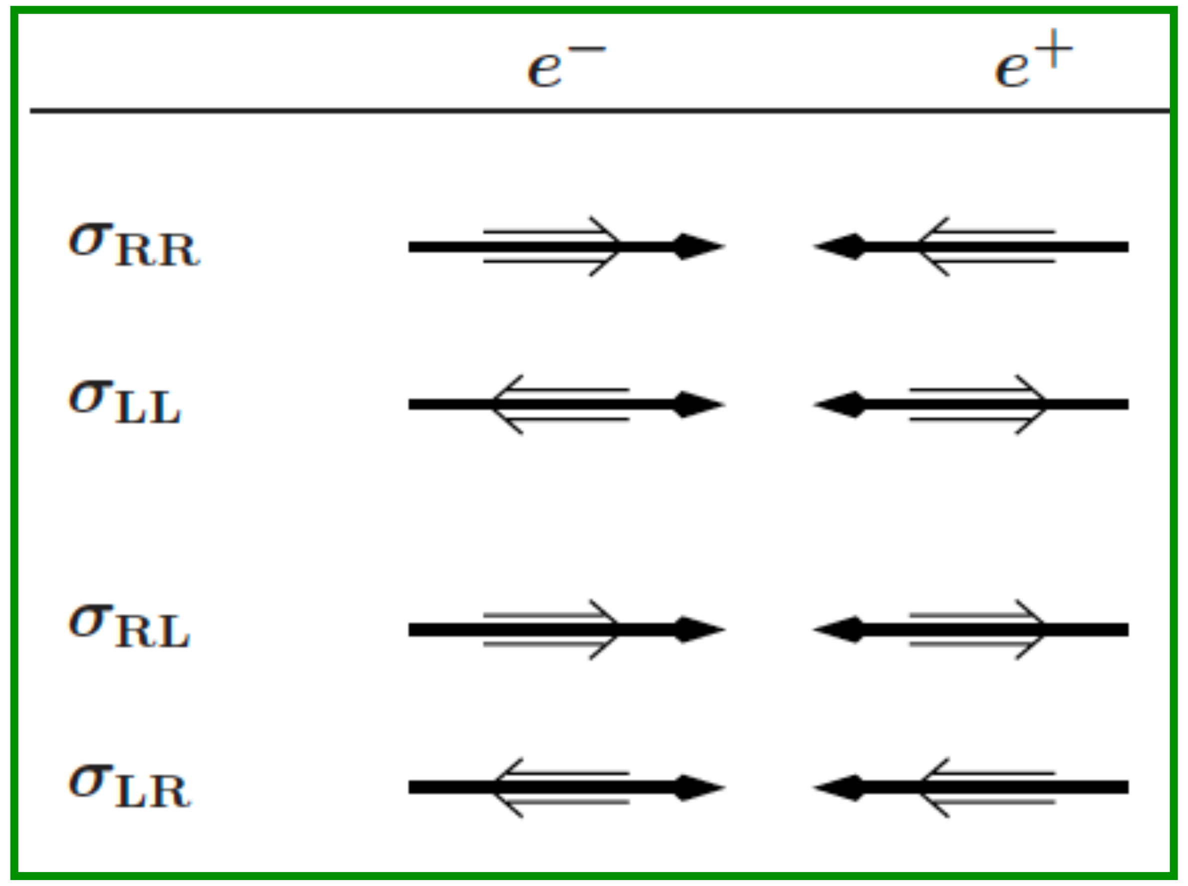
detailed reasoning c.f. arXiv:1506.07830

Future e^+e^- Colliders and (longitudinally) Polarised Beams

- Longitudinally **polarised beams** are a special feature of **Linear e^+e^- Colliders**:
  - SLC: $P(e^-) = \pm 80\%$, $P(e^+) = 0\%$
  - ILC: $P(e^-) = \pm 80\%$, $P(e^+) = \pm 30\%$ (upgrade 60%)
  - CLIC: $P(e^-) = \pm 80\%$, $P(e^+) = 0\%$

$$\boxed{P = \frac{N_R - N_L}{N_R + N_L}}$$

- Electroweak interactions highly sensitive to chirality of fermions: $SU(2)_L \times U(1)$
  - every cross section depends on beam polarisations
  - **with both its beams polarised, ILC is "four colliders in one":**



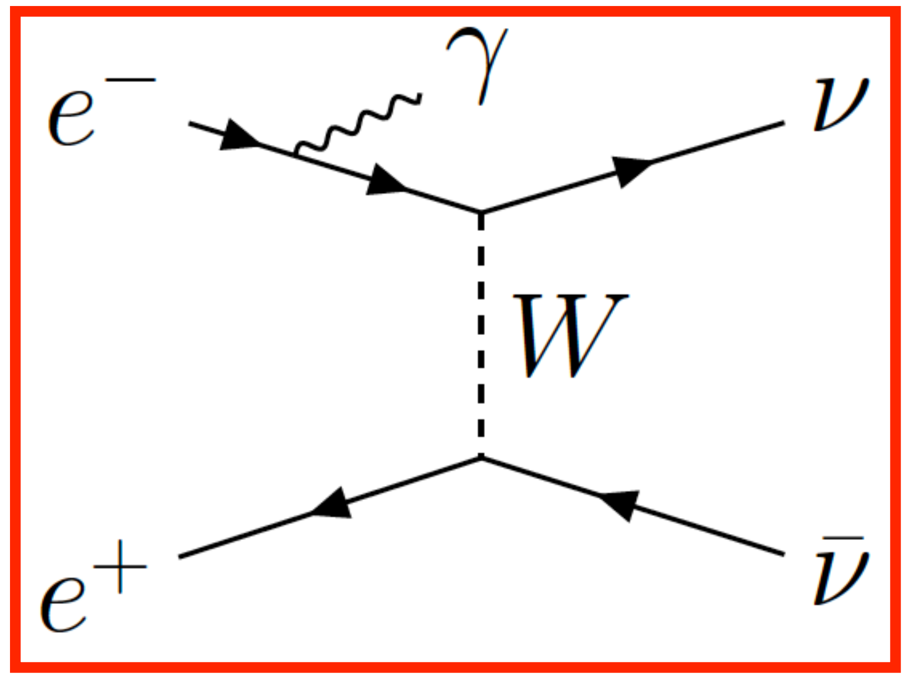
General references on polarised e^+e^- physics:

- **arXiv:1801.02840**
- **Phys. Rept. 460 (2008) 131-243**

Physics benefits of polarised beams

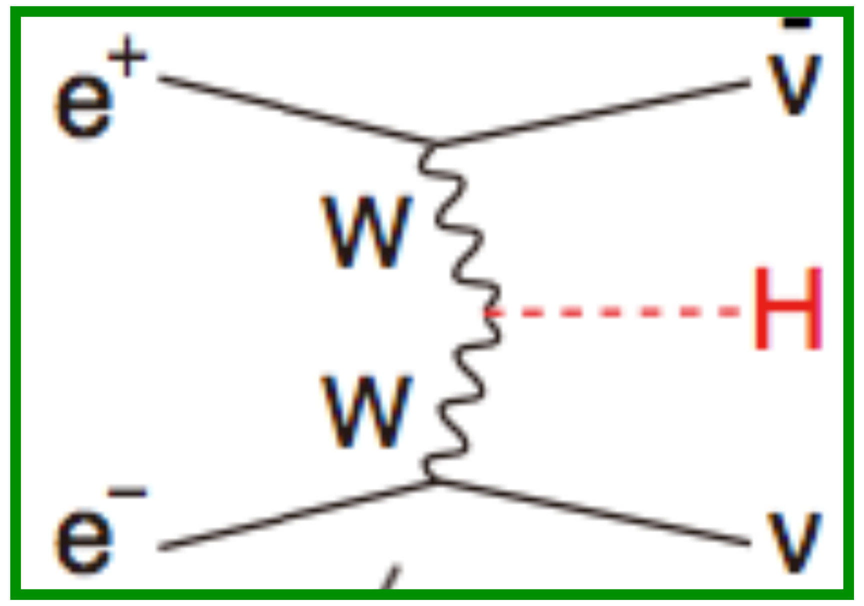
background suppression:

- $e^+e^- \to WW$ / $\nu\nu$ strongly P-dependent since t-channel only for $e^-_L e^+_R$



signal enhancement:

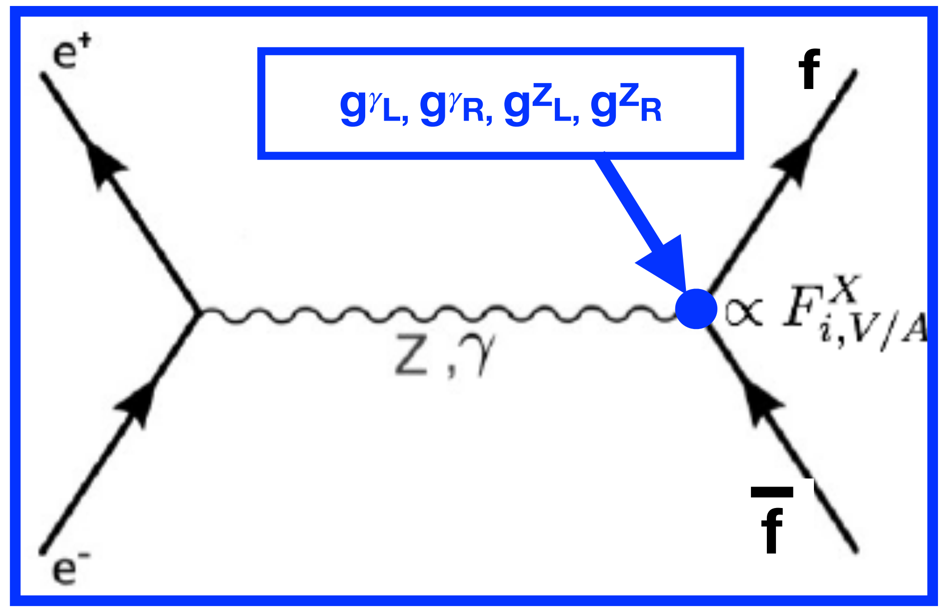
- Higgs production in WW fusion
- many BSM processes



have strong polarisation dependence => higher S/B

chiral analysis:

- SM: Z and γ differ in couplings to left- and right-handed fermions



- BSM: chiral structure unknown, needs to be determined!

redundancy & control of systematics:

- “wrong” polarisation yields “signal-free” control sample
- flipping *positron* polarisation controls nuisance effects on observables relying on *electron* polarisation
- essential: fast helicity reversal for *both* beams!

Polarised cross sections

$$\sigma_{P_{e^-}P_{e^+}} = \frac{1}{4}\{(1+P_{e^-})(1+P_{e^+})\sigma_{\mathrm{RR}} + (1-P_{e^-})(1-P_{e^+})\sigma_{\mathrm{LL}} \\ +(1+P_{e^-})(1-P_{e^+})\sigma_{\mathrm{RL}} + (1-P_{e^-})(1+P_{e^+})\sigma_{\mathrm{LR}}\}.$$

- For σ_{RR}, σ_{LR} etc, use the generator cross sections given in the event header:

```
float xsection = evt->parameters().getFloatVal("CrossSection_fb");
```

- pre-factors are the respective event weights:

```
// polarisation weights for {LR, RL, LL, RR} events, as example for target P(e-,e+)=(-80%,+30%):
// LR: polweight = (1-epol_target)*(1+ppol_target)/4.; // -80%,+30% => 1.8 * 1.3 / 4. = 0.585
// RL: polweight = (1+epol_target)*(1-ppol_target)/4.; // -80%,+30% => 0.2 * 0.7 / 4. = 0.035
// LL: polweight = (1-epol_target)*(1-ppol_target)/4.; // -80%,+30% => 1.8 * 0.7 / 4. = 0.315
// RR: polweight = (1+epol_target)*(1+ppol_target)/4.; // -80%,+30% => 0.2 * 1.3 / 4. = 0.065
```

- Note: data sets with (sign(Pe-),sign P(e+)) = (-,+) and (+,-) often have
  - different initial S/B ratio
  - different background composition => different kinematics etc

=> analyse data sets with different polarisation signs separately, different cut optimisation - either combine results afterwards, or exploit polarisation dependence in interpretation

Useful tools

- `anajob [your .slcio file]  | less`
  prints (after some header information) the list of collections available on each event, incl. their number of elements

  => try it - what do you see?

- `dumpevent [your .slcio file] [event number] | less`
  prints the content of all collections on the given event

  => try it - and find

    - the IsolatedMuons collection
    - the Refined2Jet collection

Now it is your turn !

- Try to improve the signal-to-background ratio by applying a cut on the sum of the b-likeliness values of the two jets.
- For this, read in the `Refined2Jets` collection, check that it is there and contains 2 jets.
- Then get the b-likeliness values (MVA output between 0 and 1).
- You find an example of how to access jets and b-tag information in ./examples/jet_btag.C.
- Take a look at this (of course you can also run it if you like!) and modify your `higgs_recoil_with_bkg.C` such that the recoil mass histograms are only filled if the sum of the two b-likeliness values > 1.

How to continue

- how to get more data:
  - regularly check http://ilcsnowmass.org - large data sets (SM + Higgs) will appear there soon:
    - Delphes-miniDSTs of for 250 GeV, 350 GeV, 500 GeV, 1 TeV (from ILC TDR MC production, Whizard 1.95)
    - SGV-miniDSTs of new 250 GeV Whizard 2.8.4 samples
    - ILD-miniDSTs (full simulation) of TDR MC production - priorisation depending on user requests !
  - if you need additional samples - eg BSM signals: **contact us!**
    => depending on size/ complexity of request, we'll either produce them or teach you how to produce them
- choose a topic: take a look into arXiv:2007.03650 - and don't hesitate to **contact us** if you have questions!
- further tutorials upcoming (tbd):
  - Whizard
  - delphes2lcio
  - DD4HEP / Marlin
  - e+e- analysis dos & don'ts

Contact information

- LCC Physics Working Group conveners:
  - Keisuke Fujii (keisuke.fujii@kek.jp), Christophe Grojean (christophe.grojean@desy.de), Michael Peskin (mpeskin@ slac.stanford.edu)
- ILC detector concept group physics coordinators:
  - SiD: Tim Barkow (timb@slac.stanford.edu)
  - ILD: Keisuke Fujii (keisuke.fujii@kek.jp), Jenny List (jenny.list@desy.de)
- ILC contacts for the various Energy Frontier working groups
  - EF01: Shin-ichi Kawada (shin-ichi.kawada@desy.de)
  - EF02: Maxim Perelstein (m.perelstein@cornell.edu)
  - EF03: Roman Poeschl (poeschl@lal.in2p3.fr)
  - EF04: Sunghoon Jung (sunghoonj@snu.ac.kr)
  - EF05: Juergen Reuter (juergen.reuter@desy.de)
  - EF08: Mikael Berggren (mikael.berggren@desy.de)
  - EF09: Taikan Suehara (suehara@phys.kyushu-u.ac.jp)
  - EF10: Aleksander Filip Żarnecki (Filip.Zarnecki@fuw.edu.pl)
  - TF07: Mihoko Nojiri (nojiri@post.kek.jp)
- Technical support: ilc-snowmass@slac.stanford.edu; ilc-snowmass on Slack

This tutorial:
Jan Strube: jstrube@uoregon.edu
Chris Potter: ctp@uoregon.edu
Jenny List: jenny.list@desy.de

and behind the scenes:
Norman Graf: ngraf@slac.stanford.edu
Daniel Jeans: daniel.jeans@kek.jp
Remi Ete: remi.ete@desy.de
Frank Gaede: frank.gaede@desy.de